This listing of claims will replace all prior versions, and listings, of claims in the application:

Listing of Claims

1. (Currently Amended) A method for querying instances of asset classes in a digital library coupled to a graphical user interface, comprising:

providing at least one asset class having at least one attribute, wherein each attribute is defined to have an attribute object comprising an external data object and attribute object type identifying a type of the attribute object, wherein one attribute is defined to have an attribute object comprising at least one multimedia file;

providing an asset object for each instance of one asset class, wherein the asset object includes information for the attributes in the instance of the asset class;

generating the graphical user interface including fields for receiving user input for ~~providing~~ a query indicating an asset name, search predicate, at least one attribute operator, and attribute value, wherein the attribute operator is associated with at least one attribute included in the indicated asset name; and

processing the query by accessing asset object instances of the asset name to determine asset object instances having one attribute object for the attribute associated with the attribute operator that satisfies the search predicate and the attribute value.

2. (Original) The method of claim 1, wherein the query includes multiple attribute operators and attribute values to query asset object instances whose attribute objects match the attribute values and search predicate for each attribute operator.

3. (Original) The method of claim 1, wherein each asset object instance includes information on a file location of attribute objects providing the attributes for the asset object instance, and wherein processing the query to search the attribute object comprises accessing the attribute object at the file location indicated in the asset object instance having the attribute object.

4. (Previously Presented) The method of claim 3, wherein the attribute object type indicates one of a plurality of different data structure formats that are searchable through separate application programs, and wherein processing the query to search the attribute object further comprises:

determining from the object type for the attribute associated with the attribute operator one application program capable of accessing the data structure format of the attribute object type; and

calling the determined application program to determine whether the attribute object for the asset object instance includes the attribute value in the query.

5. (Original) The method of claim 4, wherein at least one attribute object comprises a database object accessed through a database application program and at least one attribute object comprises a text object accessed through a text search engine application program.

6. (Canceled)

7. (Currently Amended)) The method of claim 1, wherein one attribute comprises a relationship attribute that defines an association of a first and second asset types and a relationship attribute object associating instances of the first and second asset types, wherein the attribute operator for a relationship attribute is associated with one attribute from the first and second asset types, wherein processing the query to search the attribute object for each asset object instance of the first asset type further comprises:

accessing the relationship attribute object to determine all asset object instances of the second asset type associated with the asset object instance; and

for each determined asset object instance, processing the query by determining the determined asset object instances of the second asset type whose attribute object for the attribute of the second asset type associated with the attribute operator matches the attribute value and satisfies the search predicate.

8. (Original) The method of claim 7, wherein the relationship attribute object comprises a database table, wherein a first column in the database table is for unique identifiers

of instances of the first asset type and a second column in the database table is for unique identifiers of instances of the second asset type, wherein a row in the database table identifies one instance of the first asset type identified by the unique identifier in the first column of the row that is associated with one instance of the second asset type identified by the unique identifier in the second column of the row.

9. (Original) The method of claim 1, wherein the definition of each attribute for an asset class is implemented in an Extensible Markup Language (XML) document, wherein each defined attribute for an asset class comprises a tagged element in the XML document and wherein information for each attribute is embedded in at least one tagged attribute of the tagged element for the attribute.

10. (Canceled)

11. (Original) The method of claim 10, further comprising:

in response to receiving a user entered asset name for the query, determining attribute operators associated with attributes of the asset name;

generating a display of a list of the determined attribute operators; and

receiving user selection of one of the determined attribute operators from the displayed list, wherein the user selected attribute operator is used in the query.

12. (Currently Amended) A system for querying data, comprising:

a digital library including asset classes;

a graphical user interface coupled to the digital library;

a computer readable medium including:

(i) a definition of at least one asset class having at least one attribute, wherein each attribute is defined to have an attribute object comprising an external data object and attribute object type identifying a type of the attribute object ~~an attribute object type~~, wherein one attribute is defined to have an attribute object comprising at least one multimedia file;

(ii) an asset object for each instance of one asset class, wherein the asset object includes information for the attributes in the instance of the asset class;

(iii) generating the graphical user interface including fields for receiving user input for a query indicating an asset name, search predicate, at least one attribute operator, and attribute value, wherein the attribute operator is associated with at least one attribute included in the indicated asset name; and

(iv) ~~means for~~ processing the query by accessing asset object instances of the asset name to determine asset object instances having one attribute object for the attribute associated with the attribute operator that satisfies the search predicate and the attribute value.

13. (Original) The system of claim 12, wherein the computer readable medium comprises a plurality of computer readable devices, wherein the computer readable devices comprise at least one of storage devices, memory devices, and transmission media.

14. (Original) The system of claim 12, wherein the query includes multiple attribute operators and attribute values to query asset object instances whose attribute objects match the attribute values and search predicate for each attribute operator.

15. (Original) The system of claim 12, wherein each asset object instance includes information on a file location of attribute objects providing the attributes for the asset object instance, and wherein the means for processing the query to search the attribute object further accesses the attribute object at the file location indicated in the asset object instance having the attribute object.

16. (Currently Amended) The system of claim 15, wherein the attribute object indicates one of a plurality of different data structure formats that are searchable through separate application programs, and wherein processing the query to search the attribute object further comprises:

means for determining from the object type for the attribute associated with the attribute operator one application program capable of accessing the data structure format of the attribute object type; and

means for calling the determined application program to determine whether the attribute object for the asset object instance includes the attribute value in the query.

17. (Original) The system of claim 16, wherein at least one attribute object comprises a database object accessed through a database application program and at least one attribute object comprises a text object accessed through a text search engine application program.

18. (Canceled)

19. (Currently Amended) The system of claim 12, wherein one attribute comprises a relationship attribute that defines an association of a first and second asset types and a relationship attribute object associating instances of the first and second asset types, wherein the attribute operator for a relationship attribute is associated with one attribute from the first and second asset types, wherein the means for processing the query to search the attribute object for each asset object instance of the first asset type performs:

accessing the relationship attribute object to determine all asset object instances of the second asset type associated with the asset object instance; and

for each determined asset object instance, processing the query by determining the determined asset object instances of the second asset type whose attribute object for the attribute of the second asset type associated with the attribute operator matches the attribute value and satisfies the search predicate.

20. (Original) The system of claim 19, wherein the relationship attribute object comprises a database table, wherein a first column in the database table is for unique identifiers of instances of the first asset type and a second column in the database table is for unique identifiers of instances of the second asset type, wherein a row in the database table identifies one instance of the first asset type identified by the unique identifier in the first column of the

row that is associated with one instance of the second asset type identified by the unique identifier in the second column of the row.

21. (Original) The system of claim 12, wherein the definition of each attribute for an asset class is implemented in an Extensible Markup Language (XML) document, wherein each defined attribute for an asset class comprises a tagged element in the XML document and wherein information for each attribute is embedded in at least one tagged attribute of the tagged element for the attribute.

22. (Canceled)

23. (Currently Amended) The system of claim 22, further comprising:

means for determining attribute operators associated with attributes of the asset name in response to receiving a user entered asset name for the query;

means [[fo]] for generating a display of a list of the determined attribute operators; and

means for receiving user selection of one of the determined attribute operators from the displayed list, wherein the user selected attribute operator is used in the query.

24. (Currently Amended) An article of manufacture including code for querying instances of asset classes in a digital library coupled to a graphical user interface, wherein the code causes operations comprising:

providing at least one asset class having at least one attribute, wherein each attribute is defined to have an attribute object comprising an external data object and attribute object type identifying a type of the attribute object, wherein one attribute is defined to have an attribute object comprising at least one multimedia file;

providing an asset object for each instance of one asset class, wherein the asset object includes information for the attributes in the instance of the asset class;

generating the graphical user interface including fields for receiving user input for ~~providing~~ a query indicating an asset name, search predicate, at least one attribute operator, and attribute value, wherein the attribute operator is associated with at least one attribute included in the indicated asset name; and

processing the query by accessing asset object instances of the asset name to determine asset object instances having one attribute object for the attribute associated with the attribute operator that satisfies the search predicate and the attribute value.

25. (Original) The article of manufacture 24, wherein the query includes multiple attribute operators and attribute values to query asset object instances whose attribute objects match the attribute values and search predicate for each attribute operator.

26. (Original) The article of manufacture 24, wherein each asset object instance includes information on a file location of attribute objects providing the attributes for the asset object instance, and wherein processing the query to search the attribute object comprises accessing the attribute object at the file location indicated in the asset object instance having the attribute object.

27. (Previously Presented) The article of manufacture 26, wherein the attribute object type indicates one of a plurality of different data structure formats that are searchable through separate application programs, and wherein processing the query to search the attribute object further comprises:

determining from the object type for the attribute associated with the attribute operator one application program capable of accessing the data structure format of the attribute object type; and

calling the determined application program to determine whether the attribute object for the asset object instance includes the attribute value in the query.

28. (Original) The article of manufacture 27, wherein at least one attribute object comprises a database object accessed through a database application program and at least one attribute object comprises a text object accessed through a text search engine application program.

29. (Canceled)

30. (Currently Amended) The article of manufacture 24, wherein one attribute comprises a relationship attribute that defines an association of a first and second asset types and a relationship attribute object associating instances of the first and second asset types, wherein the attribute operator for a relationship attribute is associated with one attribute from the first and second asset types, wherein processing the query to search the attribute object for each asset object instance of the first asset type further comprises:

accessing the relationship attribute object to determine all asset object instances of the second asset type associated with the asset object instance; and

for each determined asset object instance, processing the query by determining the determined asset object instances of the second asset type whose attribute object for the attribute of the second asset type associated with the attribute operator matches the attribute value and satisfies the search predicate.

31. (Original) The article of manufacture 30, wherein the relationship attribute object comprises a database table, wherein a first column in the database table is for unique identifiers of instances of the first asset type and a second column in the database table is for unique identifiers of instances of the second asset type, wherein a row in the database table identifies one instance of the first asset type identified by the unique identifier in the first column of the row that is associated with one instance of the second asset type identified by the unique identifier in the second column of the row.

32. (Original) The article of manufacture 24, wherein the definition of each attribute for an asset class is implemented in an Extensible Markup Language (XML) document, wherein each defined attribute for an asset class comprises a tagged element in the XML document and wherein information for each attribute is embedded in at least one tagged attribute of the tagged element for the attribute.

33. (Canceled)

34. (Original) The article of manufacture 33, further comprising:

in response to receiving a user entered asset name for the query, determining attribute operators associated with attributes of the asset name;

generating a display of a list of the determined attribute operators; and

receiving user selection of one of the determined attribute operators from the displayed list, wherein the user selected attribute operator is used in the query.